BIOINFORMATICS APPLICATIONS NOTE

Vol. 24 no. 23 2008, pages 2786–2787
doi:10.1093/bioinformatics/btn522

Genetics and population analysis

A flexible forward simulator for populations subject to selection and demography

Ryan D. Hernandez*

Biological Statistics and Computational Biology, Cornell University, Ithaca, NY 14850, USA

Received on August 13, 2008; revised on September 19, 2008; accepted on October 4, 2008

Advance Access publication October 7, 2008

Associate Editor: Alex Bateman

ABSTRACT

Summary: This article introduces a new forward population genetic simulation program that can efficiently generate samples from populations with complex demographic histories under various models of natural selection. The program (SFS_CODE) is highly flexible, allowing the user to simulate realistic genomic regions with several loci evolving according to a variety of mutation models (from simple to context-dependent), and allows for insertions and deletions. Each locus can be annotated as either coding or non-coding, sex-linked or autosomal, selected or neutral, and have an arbitrary linkage structure (from completely linked to independent).

Availability: The source code (written in the C programming language) is available at http://sfscode.sourceforge.net, and a web server (http://cbsuapps.tc.cornell.edu/sfscode.aspx) allows the user to perform simulations using the high-performance computing cluster hosted by the Cornell University Computational Biology Service Unit.

Contact: rhernandez@uchicago.edu

Supplementary information: An extensive user's manual, performance statistics, and comparisons of patterns of genetic variation generated by SFS_CODE to theoretical expectations under various non-stationary demographic histories and models of natural selection are available on the project website: http://sfscode.sourceforge.net.

1 INTRODUCTION

Forward population genetic simulations have long played a crucial role in evolutionary biology, and have been advocated nearly as long as computers have been available (Fraser, 1957). Forward simulations have been useful for guiding our intuition, testing theoretical approximations and evaluating the power of statistical tests, yet they remain an underutilized tool in current research. By following an *in silico* population generation by generation and mimicking all stages of the life cycle, it is possible to simulate data under highly complex scenarios that capture many of the factors that affect natural populations. However, with complexity generally comes a computational burden, the cost of which has driven many studies to use simplified approximations.

In contrast to forward simulations, generating samples backwards in time under the coalescent process (Hudson, 2002) can be extremely fast. However, because coalescent models of natural selection remain limited, forward simulations are the primary option for detailed analyses (particularly when considering selection across many linked sites). While previous implementations of forward simulation programs (e.g. Carvajal-Rodríguez, 2008; Guillaume and Rougemont, 2006; Hey, 2004; Padhukasahasram *et al.*, 2008; Peng and Kimmel, 2005) have produced a wide range of options geared toward mimicking natural populations, many require the user to update source code files or write extensive scripts for their use. Here, I present the program SFS_CODE (Selection on Finite Sites under Complex Demographic Events) that provides flexibility and several novel features. In addition to allowing for a context-dependent mutation model (including CpG-effects, a major driver of mammalian evolution), this program also allows the user to simulate insertions and deletions as well as evolve populations under certain models of domestication. Importantly, SFS_CODE is a self-contained program that is easily compiled, and can be run using simple command-line flags on a desktop computer or distributed across a computing cluster.

2 FEATURES AND METHODS

Among the features implemented in SFS_CODE is the ability to simulate more realistic gene regions, whereby each locus can be annotated as either coding or non-coding (e.g. exons, introns and up-/downstream regions). Mutations that fall within coding regions are classified as synonymous or non-synonymous based on the Universal Genetic Code, thereby allowing non-synonymous mutations to be driven by natural selection, while synonymous mutations can remain neutral. More generally, loci can have an arbitrary linkage structure (specifying either physical or genetic distances), and can evolve either neutrally or be subject to natural selection. Selective effects can be drawn from a wide range of possibilities (from a constant effect to a mixture of Gamma or Normal distributions with user-defined mixture components), and can vary across loci as well as over time. In addition, several mutation models have been implemented, from standard models of equal mutation rates (Jukes and Cantor, 1969) and transition-transversion biases (Kimura, 1980), to fully context-dependent models of mammalian evolution with CpG effects (Hwang and Green, 2004).

In SFS_CODE, several populations can be simulated with arbitrary divergence times and a general migration matrix (which can vary over time). Both male and female sexes are maintained, allowing biased sex ratios in each population as well as sex-biased migration patterns. Each population can experience its own demographic history (including multiple epochs of instantaneous, exponential and/or logistic population growth/decay), be

*Present address: Department of Human Genetics, University of Chicago, Chicago, IL 60637, USA.

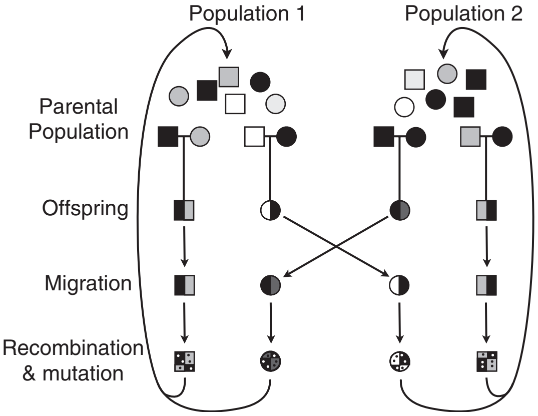

Fig. 1. The basic life cycle that takes place every generation. Each individual in the population is generated from a male and female chosen from the parental population with probabilities given by their relative fitness in their sex (only necessary for diploid and allo/autotetraploid populations, as haploid populations are simulated asexually). The new individuals are then able to migrate among populations. The gametes of the new populations then undergo recombination and mutation (both at Poisson rates) before being passed on to the next generation. Note that circles represent females, squares represent males and different colors indicate different haplotypes.

Table 1. Information for each mutation

Physical location (bp)
Frequency (sample and/or population)
Generation arose
Generation fixed
Fitness effect
Ancestral/derived nucleotides
Ancestral/derived amino acids
Flanking nucleotides[a]
Non-synonymous and/or CpG[a]
Autosomal/sex-linked
Length (if indel)
Nucleotides (if insertion)

[a]At time of mutation.

exposed to differential effects of natural selection, evolve with different generation times or go extinct at any time.

At the beginning of the simulation, there is a single DNA sequence that is carried by every individual in the population. This sequence is either drawn from the stationary distribution given by the user-specified mutation model (using a Markov chain Monte Carlo technique described in the user's manual), or supplied by the user (e.g. a real genomic sequence). A user-specified burn-in period of many generations (Fig. 1) ensues during which new mutations are introduced to allow the ancestral population to reach mutation/selection balance. Upon completion of the burn-in period, speciation events and demographic effects can occur. After the final generation is simulated, a random sample of individuals (including all of their chromosomes) is drawn without replacement.

In order to make the program as broadly useful as possible, `SFS_CODE` stores many details about each mutation event (Table 1). By modeling each locus as having finitely many sites that can receive multiple mutations (infinite-sites options also available), more realistic data can be generated to better understand the factors contributing to observed sequence data (e.g. Hernandez *et al*., 2007), and to better understand patterns of synonymous and non-synonymous variation within coding genes (e.g. Boyko *et al*., 2008). By reporting additional information about each mutation (Table 1), it is possible to use `SFS_CODE` to address a wide range of questions, from specific details regarding the relationship between the age of a mutation and its effect on fitness to general patterns of linkage disequilibrium between loci at a given physical distance, etc.

In order to improve efficiency, `SFS_CODE` only stores a single consensus sequence for each population. Unique haplotypes then only need to contain the specific mutations that define them [which are stored as a self-balancing binary tree for quick retrieval (Sleator and Tarjan, 1985)]. A given individual is then just a collection of haplotypes at each locus. Such data structures enable efficient scaling properties, such that for fixed population-scaled mutation and recombination rates of 0.001 per site, simulating 1000 diploid individuals for 10 000 generations takes 1.09 s for a 1 Kb sequence, but only 255.98 s for a 1 Mb sequence.

ACKNOWLEDGEMENTS

The author would like to Carlos D. Bustamante for the initial motivation, Ryan N. Gutenkunst for testing the program and comments on the documentation, Molly Przeworski for comments and support and the Computational Biology Service Unit from Cornell University.

Funding: National Institutes of Health (grant GM72861 to Molly Przeworski, partially).

Conflict of Interest: none declared.

REFERENCES

Boyko,A.R. *et al*. (2008) Assessing the evolutionary impact of amino acid mutations in the human genome. *PLoS Genet.*, **4**, e1000083.

Carvajal-Rodríguez,A. (2008) Genomepop: a program to simulate genomes in populations. *BMC Bioinformatics*, **9**, 223.

Fraser,A.S. (1957) Simulating of genetic systems by automatic digital computers. *Aust. J. Biol. Sci.*, **10**, 484–491.

Guillaume,F. and Rougemont,J. (2006) Nemo: an evolutionary and population genetics programming framework. *Bioinformatics*, **22**, 2556–2557.

Hernandez,R.D. *et al*. (2007) Context dependence, ancestral misidentification, and spurious signatures of natural selection. *Mol. Biol. Evol.*, **24**, 1792–1800.

Hey,J. (2004) FPG – a computer program for forward population genetic simulation. Available at http://lifesci.rutgers.edu/~heylab/HeylabSoftware.htm#FPG. (last accessed date October 21, 2008).

Hudson,R.R. (2002) Generating samples under a Wright-Fisher neutral model of genetic variation. *Bioinformatics*, **18**, 337–338.

Hwang,D.G. and Green,P. (2004) Bayesian Markov chain Monte Carlo sequence analysis reveals varying neutral substitution patterns in mammalian evolution. *Proc. Natl Acad. Sci. USA*, **101**, 13994–14001.

Jukes,T.H. and Cantor,C.R. (1969) Evolution of protein molecules. In Munro,H.N. (ed.) *Mammalian Protein Metabolism*, Academic Press, New York, pp. 21–132.

Kimura,M. (1980) A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *J. Mol. Evol.*, **16**, 111–120.

Padhukasahasram,B. *et al*. (2008) Exploring population genetic models with recombination using efficient forward-time simulations. *Genetics*, **178**, 2417–2427.

Peng,B. and Kimmel,M. (2005) simuPOP: a forward-time population genetics simulation environment. *Bioinformatics*, **21**, 3686–3687.

Sleator,D.D. and Tarjan,R.E. (1985) Self-adjusting binary search trees. *J. ACM*, **32**, 652–686.